NASA/TM—2000-209177

Cascade Optimization for Aircraft Engines With Regression and Neural Network Analysis-Approximators

Surya N. Patnaik
Ohio Aerospace Institute, Cleveland, Ohio

James D. Guptill, Dale A. Hopkins, and Thomas M. Lavelle
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center

July 2000

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076
Price Code: A03

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100
Price Code: A03

CASCADE OPTIMIZATION FOR AIRCRAFT ENGINES WITH REGRESSION AND NEURAL NETWORK ANALYSIS-APPROXIMATORS

Surya N. Patnaik
Ohio Aerospace Institute
Cleveland, Ohio 44142

James D. Guptill, Dale A. Hopkins, and Thomas M. Lavelle
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

SUMMARY

The NASA Engine Performance Program (NEPP) can configure and analyze almost any type of gas turbine engine that can be generated through the interconnection of a set of standard physical components. In addition, the code can optimize engine performance by changing adjustable variables under a set of constraints. However, for engine cycle problems at certain operating points, the NEPP code can encounter difficulties: nonconvergence in the currently implemented Powell's optimization algorithm and deficiencies in the Newton-Raphson solver during engine balancing. A project was undertaken to correct these deficiencies. Nonconvergence was avoided through a cascade optimization strategy, and deficiencies associated with engine balancing were eliminated through neural network and linear regression methods. An approximation-interspersed cascade strategy was used to optimize the engine's operation over its flight envelope. Replacement of Powell's algorithm by the cascade strategy improved the optimization segment of the NEPP code. The performance of the linear regression and neural network methods as alternative engine analyzers was found to be satisfactory. This report considers two examples—a supersonic mixed-flow turbofan engine and a subsonic waverotor-topped engine—to illustrate the results, and it discusses insights gained from the improved version of the NEPP code.

INTRODUCTION

The NASA Engine Performance Program (NEPP) is a gas-turbine engine-cycle simulation code. This code can configure and analyze almost any type of gas turbine engine that can be generated through the interconnection of a set of standard physical components: propeller, inlet, ducts, combustor, fan, compressors, turbines, shafts, heat exchangers, flow splitters, subsonic mixers and/or supersonic ejectors, nozzles and water injectors or gas generators. The engine can be designed for different types of fuels: standard hydrocarbon jet fuel and cryogenic fuel and slurries. For thermodynamic analysis, built-in curve fits can be generated from empirical data available in NEPP. For the analysis of jet and rocket fuels, an auxiliary chemical equilibrium composition model is available (ref. 1). The NEPP code has been successfully used to simulate a wide range of engines from turboshaft and turboprops to airturbo-rockets and supersonic variable-cycle engines. A description of the NEPP program, with typical input files for a set of engine configurations, is given in references 2 and 3. Since its inception (ref. 4), the NEPP program has been continuously undergoing improvement to keep pace with the advanced gas turbine engines envisioned for the 21st century. NEPP simulation has decreased engine cycle analysis time and improved engine model fidelity.

The NEPP code has a numerical optimization capability to increase engine performance. The program allows the maximization or minimization of a cost function for a set of independent variables subjected to a number of specified behavior parameters of the engine, which act as the constraints. In the NEPP code, the resulting optimization problem is solved using Powell's method (ref. 5), which was developed in the early sixties. It has been observed that for certain engine problems Powell's method can produce an overdesign condition with fewer active constraints than the correct optimum solution or can experience convergence difficulties. A project was undertaken to correct the optimization-related deficiency in the NEPP code by augmenting it with a cascade optimization strategy that was

developed recently (ref. 6). During the implementation of the new optimization algorithm for the solution of multimission air-breathing engine problems with multiple operating points within a flight envelope, difficulties were encountered in engine cycle analysis. To overcome this problem, we created approximate analyzers from the original NEPP code through the use of two competing analysis-approximators: neural network and linear regression methods. This report summarizes the results and insights gained from the improved NEPP code.

The report is organized as follows: a brief overview of the NEPP code with emphasis on engine operation optimization, two illustrative examples—a supersonic mixed-flow turbofan engine and a subsonic waverotor-topped engine, the cascade strategy with illustrations, a brief description of the neural network and regression approximations, neural network and regression solutions, and conclusions.

OPTIMIZATION CAPABILITY OF THE NASA ENGINE PERFORMANCE PROGRAM

The engine cycle simulation code NEPP is popular in U.S. aircraft engine companies. The early versions of this code (refs. 4 and 7) have been improved by introducing multiple modes of operation to simulate variable-cycle engines, "stacked" component maps for variable geometry components, and optimization capability. The current code (ref. 2) can simulate the steady-state design and off-design performance of almost any gas turbine engine. Its chemical dissociation subprogram can model different types of fuel, including standard hydrocarbon jet fuel and cryogenic fuel and slurries. In addition to modeling air-breathing propulsion engines, NEPP can simulate airturborockets, ejector mixers, and rockets.

The numerical optimization capability of the NEPP code, the subject matter of this paper, casts the engine operation as a standard nonlinear mathematical programming problem:

Find the design variables $\{D\}$
To optimize a cost function C_f
Subjected to a set of inequality constraints $\{g\}$

Variable-cycle engines have to perform satisfactorily over their flight envelopes, which consist of a number of operating points defined by altitude, Mach number, and power setting combinations. Engine operation design becomes a sequence of interdependent optimization subproblems, one for each operating point. The optimization process adjusts a few engine parameters. The difficulty in the optimization problem does not lie with the number of active design variables, but rather with its multiple operating-point character, constraint validity ranges, and the iterative nature of engine cycle analysis. In the original NEPP code, subproblems were solved in sequence using Powell's algorithm. This algorithm of the sixties has been replaced by a state-of-the-art cascade optimization strategy.

During engine optimization, an anomaly associated with the Newton-Raphson solver was observed in the analysis portion of the NEPP code. In the NEPP code, an engine is balanced by varying independent parameters until dependent parameters are matched. For example, the compressor operating point must be varied until the compressor flow error is reduced to zero. In addition, the shaft speed must be varied until the amount of power being supplied by the turbine matches the amount of power being used by the compressor. The Newton-Raphson solver (ref. 8), which does this balancing, utilizes the solution of the previous operating point as the initial guess for the current operating point. During an attempt to balance an engine, this solver converges for small changes in operating conditions or when the starting point is close to the solution point. The solver encounters convergence difficulties when there are relatively large jumps in operating conditions. In optimization, such a deficiency becomes troublesome especially during one-dimensional searches, when the feasible design space can be violated. In most situations, the algorithm returns to the feasible region to continue optimization calculations. However, the engine analyzer is unable to return to the feasible region when operating characteristics change abruptly, leading to a termination of the optimization process. The authors attempted to overcome this problem by approximating the engine analysis. Two competing models—neural network and regression methods—provide analysis-approximation for the engines. An engine design optimization problem, in other words, can be solved using three analyzers: the original NEPP analyzer, the derived analyzer using neural network approximations, and the linear-regression-based analyzer. The NEPP code with the cascade strategy coupled with neural network and regression approximations has been used successfully to solve both supersonic and subsonic engine operation problems.

BEHAVIOR PARAMETERS FOR ENGINE OPTIMIZATION

NEPP provides a number of behavior parameters that can be used in engine operation optimization. Because there are a large number of parameters, we provide only a partial list of possible design variables, cost functions, and constraints.

Design Variables

The following parameters can be used as design variables for engine operation optimization:

D_1 Rotational speed of the engine shaft that provides connections between compressors, turbines, propellers, and other components
D_2 Tip speed of the propeller, when such a component is used in the engine
D_3 Waverotor speed, when such a component is used in the engine
D_4 Ratio of compressor bleed flow to total flow into the compressor
D_5 Compressor pressure ratio, its adiabatic efficiency, and its bleed
These parameters and the ratio of compressor bleed flow to total flow into the compressor can also be used for turbines and fans.
D_6 Three-dimensional map value for the fan
D_7 Flow area for the nozzle, or more specifically, the exit area for the convergent and throat area for convergent-divergent nozzle types
D_8 Ratio of the exit to the throat area for convergent-divergent nozzles
D_9 Geometrical nozzle parameters, such as length and divergence angle
D_{10} Nozzle exit static pressure
D_{11} Geometrical parameters of the duct and burner
D_{12} Ratio of the entrance and exit bleed flow to the total flow for the duct
D_{13} For splitters, the bypass ratio in each branch
D_{14} Primary flow temperature change in the heat exchanger
D_{15} Heat added to the waverotor
D_{16} Primary and secondary flow area for mixers and ejectors

The list of design variables can be expanded for other engine components.

Cost Function and Constraints

For operation optimization, several engine parameters can be considered either as behavior constraints or as components of the cost function. Some behavior parameters that can form cost components follow:

Net engine thrust is a typical cost function. This parameter can be modified to account for drag from the installation, inlet, nozzle boattail, and other protuberances. The shaft and propeller horsepower are related cost components.

C_1 Fuel flow per unit time, or a combination of thrust and fuel flow (for example, the ratio of fuel flow to the engine thrust) can become a cost component. A constraint can also be specified on fuel flow.
C_2 The NO_x emission index (as the ratio of NO_x in grams to fuel in kilograms) can be a cost component. This variable can also become a constraint with a specified limit.
C_3 For the purpose of optimization, a composite merit function can be generated as a linear combination of the component costs as

$$C_f = \sum_k w_k \gamma_k \tag{1}$$

where C_f is the cost function, w_k is a weight factor, and γ_k is the cost for the kth component.

Behavior constraints can be imposed on the following engine parameters:

g_1 Pressure ratio, temperature, and drag for inlets
g_2 Surge margin and pressure ratio for fans, compressors, and turbines
g_3 Exit temperature and corrected speed for fans and compressors
g_4 Flow factor for turbines
g_5 Jet velocity, overall pressure ratio, and static and critical pressure ratio at the throat for nozzles
g_6 Pressure, temperature, Mach number, and fuel-to-air ratio at specific flow stations along the length of the engine
g_7 Mass flow rate at the entrance or exit of the inlets
g_8 Bypass ratio, total pressure losses, and total weight flow for flow splitters
g_9 Total pressure drop, fuel mass flow, outlet temperature, and efficiency for burners

The constraint set has to be modified for waverotor augmentation.

ILLUSTRATIVE EXAMPLES

Two engine problems are used to illustrate the cascade optimization strategy and analysis approximations. Engine 1 is a supersonic mixed-flow turbofan engine for the High-Speed Civil Transport system. Engine 2 is a subsonic waverotor-topped engine.

Engine 1: Supersonic Mixed-Flow Turbofan Engine

The supersonic mixed-flow turbofan (MFTF) engine with bypass flow is configured with 15 components. The components are on 2 shafts with 17 flow stations: an inlet, a fan, a flow splitter, two ducts, a compressor, another duct, a burner, a high-pressure turbine, a low-pressure turbine, one more duct, a flow mixer, an afterburner nozzle, and so forth. The fan and low-pressure turbine are mounted on the first shaft. The second shaft carries the compressor, the high-pressure turbine, and a load. The engine was designed for a flight envelope with 122 operating points. The altitude of these operating points varies between sea level and 80 000 ft, while the speed changes between 0.0 and 2.4 Mach as shown in figure 1(a). The design objective at each operating point is to maximize the net thrust of the supersonic engine, accounting for the installation drag. The three active, independent design variables of the MFTF engine for these off-design points are—

D_1 Bypass flow ratio of the splitter between flow stations 3 and 4 (flow station 3 is located downstream of the fan, and flow station 4 is located upstream of a duct leading to the compressor)
D_2 Fan speed
D_3 Fan surge margin

Two constraints, an upper and a lower bound constraint, were specified on the following parameters:

g_1 Inlet airflow ratio
g_2 Bypass ratio in the flow splitter
g_3 Burner exit temperature exclusive of nonheated air
g_4 Compressor exit temperature
g_5 High-pressure turbine flow scale factor
g_6 Corrected speed ratio for the fan
g_7 Corrected speed ratio for the high-pressure compressor
g_8 Mixer primary entrance Mach number at flow station 11 (flow station 11 is located upstream of a flow mixer leading through a duct to the nozzle)
g_9 Mixer secondary entrance Mach number at flow station 16 (flow station 16 is located upstream of the flow mixer)
g_{10} R-value for the fan
g_{11} R-value for the high-pressure compressor

Engine 2: Subsonic Waverotor-Topped Engine

A high bypass-ratio subsonic waverotor-enhanced turbofan engine, depicted in figure 2, is considered as the second example. This engine is made of 16 components on 2 shafts with 21 flow stations. It is modeled with standard components that include an inlet and a splitter, which branch off to a compressor, a duct, and a nozzle. The main flow proceeds through a fan, a duct, a high-pressure compressor, another duct, a high-pressure turbine, a low-pressure turbine, one more duct, and a nozzle. The components mounted on the first shaft include the fan, the compressor along the secondary flow branch, the low-pressure turbine, and a load. The second shaft carries the high-pressure compressor along the main flow, the high-pressure turbine, and another load. The four-port waverotor (with its burner inlet and exhaust, compressor inlet, and turbine exhaust), is located between the high-pressure compressor and the high-pressure turbine. As depicted in figure 1(b), the engine operating condition is specified by a flight envelope with 47 operating points, with altitudes between sea level and 40 000 ft and speeds between Mach 0.0 and 0.85.

To examine the benefits that accrue from the waverotor enhancement, we optimized the engine by considering several baseline variables to be passive. The operation design objective was to maximize the net engine thrust for the following two waverotor active design variables:

D_1 Heat added to the waverotor in the range 93 700 to 131 300 Btu/sec
D_2 Waverotor speed in the range 4940 to 7660 rpm

An upper and a lower bound constraint were specified on each of the following engine parameters:

g_1 Corrected speed ratio for the compressor along the secondary flow branch in the range 0.7 to 1.01
g_2 Corrected speed ratio for the fan in the range 0.7 to 1.01
g_3 Corrected speed ratio for the high-pressure compressor along the main flow in the range 0.7 to 1.01
g_4 Waverotor unmixed temperature in the range 2000 to 3200 °R
g_5 Surge margin on the compressor along the secondary flow branch in the range 15 to 100
g_6 Surge margin on the fan in the range 15 to 100
g_7 Surge margin on the high-pressure compressor along the main flow in the range 15 to 30
g_8 Pressure ratio for the high-pressure turbine in the range 0.0 to 6.59

CASCADE OPTIMIZATION STRATEGY

Nonlinear programming optimization algorithms play an important role in the optimization of aircraft, their engines and structures, and other engineering problems. During the past few decades, several algorithms with associated computer codes have been developed. The performance of 10 different algorithms was evaluated for a set of 40 structural design problems with 10 to 50 design variables and a few hundred constraints. We observed that none of the algorithms could solve all the problems, even though most algorithms succeeded in solving at least one-third of them (refs. 9 and 10). The algorithms were used next to solve aircraft system design and variable-cycle engine problems. Even the most robust algorithm encountered difficulties in generating optimum solutions for the aircraft and engine problems. This can be attributed to diverse design variables and distorted design spaces. The aircraft problem, for example, required combining different types of design variables such as the wing area, engine thrust, temperature, and pressure ratio. The constraints (such as the takeoff and landing field lengths, compressor temperature, jet velocity, and climb thrust), which differed in magnitude and units of measure, distorted the feasible design space. The complexity was further compounded by the large sequence of implicit optimization subproblems that had to be solved to design a variable-cycle engine for multiple operating points.

Improving the two key ingredients common to most algorithms—the search direction and step length—and thereby developing a superior optimizer was seriously considered, but ruled out, because we believed that such aspects had been considered by the combined efforts of the developers of the algorithms (refs. 11 to 15). Instead, a strategy (ref. 6) that would benefit from the strength of more than one optimizer was conceived. This strategy uses a number of optimization algorithms, one at a time, in a specified sequence (see fig. 3(a)). The problem is solved by Optimizer 1, and an intermediate optimum solution is obtained. The second solution cycle is begun from the Optimizer 1 solution with some pseudorandom damping. The process is continued with the third optimizer. This

cascade strategy, which derives strength from three optimizers, was found to be superior to all three of its component optimizers.

The cascade strategy is illustrated for a subsonic aircraft system design and the two engine problems: the MFTF engine (Engine 1) and the subsonic waverotor-topped engine (Engine 2). For the subsonic aircraft system, a three-optimizer cascade was used along with the aircraft system analyzer of the Langley Research Center's Flight Optimization System code (FLOPS, ref. 16). For this problem, aircraft weight was considered as the merit function. Design variables included the wing area, wing sweep, wing aspect ratio, wing taper ratio, wing thickness-to-chord ratio, and engine thrust. The important behavior constraints were specified on the approach velocity, jet velocity, takeoff and landing field lengths, and missed-approach thrust.

The cascade strategy used a sequence of three algorithms: nonlinear quadratic programming (NLPQ, ref. 11) was followed by the method of feasible directions (FD, ref. 12); then NLPQ was used again to solve the subsonic aircraft system design problem. Figure 3(b) depicts the cascade solution, along with solutions obtained from individual algorithms. The NLPQ method, used alone, converged to a heavy solution for the aircraft weight—202 005 lb (see the insert in fig. 3(b)). Furthermore, this converged solution was infeasible (i.e., it violated constraints). Likewise, the FD algorithm alone produced a heavy design of 202 854 lb (see the insert in fig. 3(b)). In other words, neither the NLPQ nor the FD algorithm alone could successfully solve the subsonic aircraft system design problem. However, the NLPQ–FD–NLPQ cascade created from the same two optimizers successfully solved the problem with a feasible optimum solution at 199 818 lb for the aircraft weight (see fig. 3(b)). The cascade strategy reduced the aircraft weight by 1.5 percent, eliminating infeasibility and overdesign conditions encountered by the individual cascade components. For the subsonic aircraft problem, the cascade strategy was successful. In a cascade (for example, NLPQ–FD–NLPQ), the same optimizer, NLPQ, can be used more than once, but we recommend that optimizer NLPQ be separated by another optimizer, such as FD.

The solution to the MFTF engine problem utilized a two-optimizer cascade: sequential linear programming (SLP, ref. 12), followed by FD. This cascade algorithm converged to a feasible solution at 26 946 lb for the engine thrust (see fig. 3(c)). The first algorithm, SLP, produced an infeasible underdesign at 22 699 lb for optimum thrust, and it violated 11 constraints. The second algorithm, FD, reached the optimum with a thrust value of 26 946 lb, which is feasible. The cascade strategy improved the engine thrust by 19 percent and eliminated design infeasibility. The SLP–FD cascade algorithm, therefore, successfully solved the MFTF problem.

The solution to the waverotor-enhanced subsonic engine (fig. 3(d)) utilized a three-optimizer cascade: NLPQ followed by FD and NLPQ. The first NLPQ algorithm produced an infeasible design at 73 293 lb for the optimum thrust. The intermediate algorithm also produced an infeasible design condition (73 694 lb). The final NLPQ algorithm reached the feasible optimum thrust at 72 989 lb. In summary, the cascade strategy successfully solved the subsonic aircraft problem and the two engine problems, even though its components encountered difficulties when used singly.

This cascade strategy is problem dependent. Algorithm sequencing depends on the experience of the designer. Important variables encompass the sequencing of optimizers, individual stop criteria, and the damping between optimizers. Careful selection of these variables can lead to a successful, robust, and numerically efficient cascade algorithm (ref. 6). The cascade sequence can bypass mild local optima. There is no guarantee that algorithms arranged in different sequences will reach the same optimum solution (see fig. 4).

SOLUTIONS OF THE MIXED-FLOW TURBOFAN AND WAVEROTOR-TOPPED ENGINES

Solutions for the supersonic mixed-flow turbofan and subsonic waverotor-topped engines obtained using the cascade strategies described earlier (see fig. 2) and the original NEPP code (which used Powell's method) are depicted in tables I and II, respectively, for a few representative operating points. Figure 4 shows normalized solutions for both engines throughout their respective flight envelopes. The vertical axis in figure 4 represents an efficiency factor η (the ratio of NEPP solutions to the cascade results) for each operating point. A value greater than unity for this factor ($\eta > 1$) represents better performance for the original NEPP code than for the cascade strategy. Likewise, a factor less than unity ($\eta < 1$) represents superior performance for the cascade strategy. From tables I and II and figure 4, it can be observed that the cascade solution produced higher thrust ($\eta < 1$) for most of the

operating points. For a few operating points, the two solutions obtained using the original and improved versions of the NEPP code agreed. The performance improvement can become significant if the design points with increased thrust are used to size the engines.

ENGINE ANALYSIS AND OPTIMIZATION WITH NEURAL NETWORK AND REGRESSION APPROXIMATIONS

Two competing techniques—linear regression and neural networks—have been used to create approximate analyzers for any type of engine that can be modeled by the original NEPP code. The following steps were used in developing the approximate analyzers:

(1) Select the basis functions.
(2) Establish a benchmark solution.
(3) Generate input-output pairs.
(4) Train the approximate methods.
(5) Measure performance during optimization.

The MFTF and waverotor-topped engines required independent approximations for the cost function, as well as all the constraints. For each method, the MFTF engine with 22 constraints and 122 operating points required the development of 2806 independent functions. Likewise, the other engine required 799 functions for each method. Creating approximate models for engine problems requires the manipulation and management of an enormous amount of numerical data. To illustrate the approximate concepts in analysis and optimization while keeping numerical calculations within a manageable level, we selected the second problem—a waverotor-augmented subsonic engine with 10 operating points. For both the regression and neural network methods, this problem only required the generation of 170 approximate functions. The regression and neural network models are described next.

Linear Regression Analysis

Linear regression analysis can employ the following basis functions:

(1) A full cubic polynomial
(2) A quadratic polynomial
(3) A linear polynomial in reciprocal variables
(4) A quadratic polynomial in reciprocal variables
(5) Combinations of these items

Consider, for example, the regression analysis model for an n variable with a cubic polynomial in the design variables and a quadratic polynomial in the reciprocal design variables. The regression function has the following explicit form:

$$y(\vec{x}) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} \beta_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=j}^{n} \beta_{ijk} x_i x_j x_k + \sum_{i=1}^{n} \bar{\beta}_i \frac{1}{x_i} + \sum_{i=1}^{n}\sum_{j=1}^{n} \bar{\beta}_{ij} \frac{1}{x_i x_j} \tag{2}$$

where y is the function to be approximated, $\vec{x}$ is the vector of independent variables, and the regression coefficients $\{\beta\}$ are determined by using the linear least squares method incorporated in the DGELS routine of the Lapack library (ref. 17).

The gradient matrix of the regression function with respect to the design variables is obtained in closed form. For the example with n variables, the gradient matrix for the regression function has the following form:

$$\nabla y = \begin{Bmatrix} \dfrac{\partial}{\partial x_1} \\ \dfrac{\partial}{\partial x_2} \\ \vdots \\ \dfrac{\partial}{\partial x_n} \end{Bmatrix} \tag{3}$$

where

$$\frac{\partial y}{\partial x_i} = \beta_i + \sum_{i=1}^{n} \beta_{ii} x_i + \beta_{ii} x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \beta_{ij} x_i x_j + \sum_{i=1}^{n} \beta_{ij} x_i^2 + \sum_{i=1}^{n} \beta_{iii} x_i x_i + \beta_{iii} x_i^2 - \frac{\overline{\beta}_i}{x_i^2} - \frac{1}{x_i^2} \sum_{i=1}^{n} \overline{\beta}_{ii} \frac{1}{x_i} - \frac{\overline{\beta}_{ii}}{x_i^3} \tag{4}$$

Once the regression coefficients have been obtained from a single training cycle, the reanalysis and sensitivity analyses represented by equations (2) to (4) require trivial computational effort. In regression analysis, the accuracy for the approximate function and its gradient can differ significantly near and beyond the training domain.

Neural Network Approximations

The neural network approximation available for engine optimization is referred to as Cometnet. It is a general-purpose, object-oriented library. Cometnet approximates the function with the following set of kernel functions:

$$y(\vec{x}) = \sum_{r=1}^{N} \sum_{i=1}^{n_r} w_{ri} \varphi_{ri}(\vec{x}) \tag{5a}$$

$$\frac{\partial y(\vec{x})}{\partial \vec{x}_i} = \sum_{r=1}^{N} \sum_{i=1}^{n_r} w_{ri} \frac{\partial \varphi_{ri}(\vec{x})}{\partial \vec{x}_i} \tag{5b}$$

where φ_{ri} represents the N kernel functions, n_r represents the number of basis functions in a given kernel, and w_{ri} represent the weight factors.

Cometnet permits approximations with linear, reciprocal, and polynomial, as well as Cauchy and Gaussian radial, functions. A singular value decomposition algorithm (ref. 18) is used to calculate the weight factors in the approximate function during network training. A clustering algorithm in conjunction with a competing complexity-based regularization algorithm (ref. 19) is used to select suitable parameters for defining the radial functions.

Generation of a Benchmark Solution for the Subsonic Waverotor-Topped Engine

Depending on the choice of the initial design, the nonlinear engine problem exhibited some variations in the optimum solutions at different operating points. To quantify the variation in the optimum thrust, we solved the problem using the original NEPP analyzer for 10 operating points, each using 10 different initial designs. The results are given in columns 2 and 3 of table III and figure 5. The average solution for the 10 operating points is considered as the benchmark solution. The solution for thrust shows a maximum standard deviation of 1.2 percent for operating point 7 (see column 3 of table III). Modest variation was observed for the other nine operating points. The benchmark solution of unity is represented by the solid horizontal line in figure 5. The height of the vertical lines in figure 5 represent standard deviation at each operating point.

Strategy to Generate the Input-Output Training Pairs

To generate the input-output pairs to train the approximate methods, we selected 3000 design points at random within their upper and lower bounds. The NEPP code was executed for the waverotor-topped engine analysis for 10 operating points for the 3000 design points. Midway through the first operating point, the NEPP analyzer encountered convergence difficulties—producing Not-A-Numbers (NaNs) followed by premature termination. For the 1500 points for which numerical solutions could be obtained, about 10 percent, or 150 cases, encountered convergence difficulties, producing zero thrust conditions. For example, the design point with 112 350 Btu heat added and a speed of 3740 rpm produced a thrust of 56 805 lb. However, 103 080 Btu and 4826 rpm produced zero thrust instead of the correct thrust of 58 998 lb. Likewise, 113 000 Btu and 5505 rpm produced zero thrust instead of the correct 64 682 lb. Because we suspected that this behavior was associated with the step size between neighboring design points, the 3000 design points were rearranged in an ascending order for each variable to obtain two sets of design points. Input-output pairs were generated for both sets. A point-by-point comparison was made between the output results obtained for the two sets. The following criteria were used to select good quality input-output pairs.

(1) Pairs with zero thrust conditions were excluded.
(2) Pairs with thrust differences exceeding 10 lb were excluded.
(3) Pairs with zero values for constraints were excluded.
(4) Pairs with constraint differences exceeding 6 percent were excluded.

The selection process produced a set of good quality input-output pairs for the 10 operating points as shown in table IV. On an average, about 10 percent of the generated input-output pairs were excluded. The rejected pairs could have adversely affected the optimization process when the original NEPP analyzer was used.

TRAINING THE APPROXIMATE METHODS

Both regression and neural network methods were trained for the 10 operating points for the good-quality training pairs shown in table IV. For regression approximations, cubic polynomials were used in design variables and quadratic polynomials were used in reciprocal design variables. The regression coefficients were determined with the linear least square routine DGELS from the Lapack subroutine library (ref. 17). Once the coefficients were obtained, equation (2) was used for functional approximations and equations (3) and (4) for gradient calculations. Likewise, a gaussian radial function was used to generate neural network models for each operating point. For each of the 10 operating points, the process generated 17 regression and 17 neural network models (for 16 constraints and a cost function). A total of 340 approximate models were obtained for the 10 operating points. The quality of the reanalysis models was verified for a set of 100 random points by generating NEPP, regression, and neural network solutions. To quantify the level of accuracy, the following error norms were used for constraints and the objective function.

Average error in the cost function for the neural network method was defined as

$$\varepsilon_{cf} = \frac{\sum_{i=1}^{n} \left| \left(\frac{C_f^{NN} - C_f^{NEPP}}{C_f^{NEPP}} \right)_i \right|}{n} \tag{6a}$$

where ε_{cf} represents the average relative error, n is the total number of sample data points, and C_f^{NN} and C_f^{NEPP} are the values of the cost function from the neural network calculation and the original NEPP methods, respectively.

Average error in a constraint for the neural network method is defined as

$$\varepsilon_g = \frac{\sum_{i=1}^{n} \left| \left(g^{NN} - g^{NEPP} \right)_i \right|}{n} \tag{6b}$$

where ε_i represents the average error, n is the number of sample data points, and g^{NN} and g^{NEPP} are the values of the constraint from the neural network calculation and the original NEPP methods, respectively. Since the constraints g represent normalized values, the error calculation is not normalized further. Likewise, average errors can be defined for the regression methods through equations (6a) and (6b).

Table V gives the average errors for a few typical engine parameters from both methods for each of the 10 operating points. For the neural network method, the maximum error in the thrust over the 10 operating points was about 0.25 percent. For the regression method, the same parameters produced an error of less than 0.04 percent. Both neural network and regression method performance for thrust estimation can be considered adequate. Over all the constraints, the compressor surge margin (g_9) exhibited the highest error, which was 1.25 percent for the regression method versus 1.05 percent for the neural network method. For the approximation of the cost function and the constraints for all 10 operating points of the subsonic waverotor-topped engine, both neural network and regression methods trained satisfactorily, and at about the same level.

Optimization Using the Approximate Analyzers

The waverotor-topped engine was optimized utilizing both neural network and regression approximations, along with three different cascade strategies. The first cascade (FD–SUMT–NLPQ) used three algorithms in sequence: FD, the sequence of unconstrained minimizations technique (SUMT, ref. 15), and quadratic programming (NLPQ). The second cascade (NLPQ–FD–NLPQ) was created using two algorithms: NLPQ, FD, and NLPQ again. The third strategy is referred to as the approximation-interspersed cascade strategy (NLPQ/NN–FD/Regression-NLPQ/NEPP). Here, the engine was optimized first by the NLPQ algorithm with the neural network analyzer, followed by optimization using the FD algorithm with the regression model, and finally, by the NLPQ algorithm with the original NEPP analyzer.

In figure 5, the optimum thrust for each operating point obtained using the first cascade with the two approximate analyzers is represented by a triangle for regression and a circle for the neural network. The solutions obtained from both approximate methods lie within one standard deviation of the benchmark solution for each operating point (shown as the horizontal line at a normalized thrust of 1.0 in fig. 5). Both of the approximate methods performed satisfactorily and at about the same level.

For the sixth operating point, optimum solutions obtained with and without the use of cascade strategies are depicted, along with the benchmark solution, in a bar chart (see fig. 6). The solution for this operating point obtained using the approximate methods with no cascade strategy indicated a variation of 12.4 and 5.5 percent for thrust, of 16.9 and 5.8 percent for the added heat, and of 1.7 and 26.8 percent for the waverotor speed. The neural network with the NLPQ optimizer produced an infeasible design (constraint g_{14} was violated) with a thrust that was 5.5 percent lower than the benchmark thrust. Similarly, the regression model with the NLPQ optimizer produced an infeasible design with a thrust that was 12.4 percent higher than the benchmark thrust. All three cascade solutions with the approximate analyzers agreed with the benchmark solution, with minor deviations. In other words, successful optimization of the subsonic waverotor-topped engine required the cascade strategy even for the neural network and regression approximate methods.

CPU Time Estimation

The engine operation optimization is not computation intensive. Furthermore, estimation of computation time has become less significant because the cost of computation has been continuously decreasing over the past few years. However, a time estimation is provided for completeness.

The numerical calculations reported in this paper utilized three different machines for convenience: International Business Machines RS6000 (for the generation of input-output pairs and operation optimization), Silicon Graphics Indigo2 (for training the regression scheme), and Silicon Graphics Power Series 480–VGX (for training the neural network). The CPU (central processing unit) times for the optimization using the original NEPP code with three different algorithms (NLPQ alone, Cascade 1, and Cascade 2) were 22 sec, 85 sec, and 116 sec, respectively. The input-output pair generation to train the approximate analyzers consumed the bulk of the CPU time at 16 535 sec (4 hr, 35 min, 35 sec). (The time required does not account for the many unsuccessful attempts to generate good-quality input-output pairs). Regression training required a trivial amount of CPU time at 14 sec. The neural network training time was 3914 sec (1 hr, 5 min, 14 sec). Optimization run times with the regression analyzer for the three different cascade strategies were 4 sec, 15 sec, and 11 sec, respectively. Optimization with the neural network required more time than the regression analyzer. The time for optimization using the neural network is not comparable to the solution time by the regression method because several other factors influenced its execution time. This execution time is not reported.

CONCLUSIONS

Replacement of Powell's optimization algorithm by the cascade strategy improved the performance of the optimization segment of the NEPP code. The maximum improvement in the thrust was 8 percent for the MFTF engine and 5 percent for the subsonic waverotor-topped engine.

The performance of the linear regression and neural network methods as alternate engine analyzers was found to be satisfactory for the analysis and operation optimization of air-breathing propulsion engines. Both linear regression and neural networks performed at about the same level.

The engine optimization required the cascade strategy for solution using the original NEPP analyzer, as well as the neural network and regression method-based analysis-approximators.

Air-breathing propulsion engines can be optimized using either the original NEPP code or an approximate analyzer and any of three methods: (1) a single state-of-the-art optimization algorithm, (2) cascade strategies, or (3) approximation-interspersed cascade strategies (i.e., a cascade strategy that includes approximate analyzers in the first few elements of the cascade and the original NEPP analyzer in the last element).

REFERENCES

1. Gordon, S.; and McBride, B.J.: Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations. NASA SP–273, 1976.
2. Klann, J.L.; and Snyder, C.A.: NEPP Programmers Manual, Vol. 1: Technical Description and Vol. 2—Source Code Listing. NASA TM–106575–VOL–1 and NASA TM–106575–VOL–2, 1994. (Available from the NASA Glenn Aerospace Analysis Office.)
3. Plencner, R.M.; and Snyder, C.A.: The Navy/NASA Engine Program (NNEP89)—A User's Manual. NASA TM–105186, 1991.
4. Caddy, M.J.; and Shapiro, S.R.: NEPCOMP—The Navy Engine Performance Computer Program, Version I, Final Report. NADC–74045–30, Apr. 1975. (Available from the Naval Air Development Center—Air Vehicle Technology Department, Warminster, PA.)
5. Powell, M.J.D.: Efficient Method for Finding the Minimum of Function of Several Variables Without Calculating Derivatives. Computer J., vol. 7, no. 2, July 1964, pp. 155–162.

6. Patnaik, S.N.; Coroneos, R.M.; and Hopkins, D.A.: Cascade Optimization Strategy for Solution of Difficult Design Problems. Int. J. Numer. Meth. Engrg., vol. 40, no. 12, 1997, pp. 2257–2266.
7. Fishbach, L.H.; and Caddy, M.J.: NNEP: The Navy NASA Engine Program. NASA TM–X–71857, 1975.
8. Pennington, R.H.: Introductory Computer Methods and Numerical Analysis, Second Edition. The MacMillan Company, New York, 1970.
9. Patnaik, S.N., et al.: Comparative Evaluation of Different Optimization Algorithms for Structural Design Applications. Int. J. Numer. Meth. Eng. (Also, NASA CR–204606, 1997), vol. 39, Jan. 1996, pp. 1761–1774.
10. Guptill, J.D., et al.: CometBoards Users Manual: Release 1.0. NASA TM–4537, 1996. (Available online: http://gltrs.grc.nasa.gov/cgi-bin/GLTRS/browse.pl?/1996/ TM–4537.html)
11. Schittkowski, K.: NLPQL: A Fortran Subroutine for Solving Constrained Nonlinear Programming Problems. Annals of Operations Research, Vol. 5, 1986, pp. 485–500.
12. Vanderplaats, G.N.: DOT User's Manual, Version 2.00. VMA Engineering, Santa Barbara, CA, 1989.
13. Arora, J.S.: IDESIGN (SQP) User's Manual, Version 3.5.2. Optimal Design Lab., Univ. Iowa, Iowa City, IA, 1989.
14. NAG Introductory Guide—MARK 16: E04UCF, NAG FORTRAN Library Routine Document, Downers Grove, IL, 1993.
15. Miura, H.; and Schmit, L.A., Jr.: NEWSUMT: A FORTRAN Program for Inequality Constrained Function Minimization, Users' Guide. NASA CR–159070, 1979.
16. McCullers, L.A.: Aircraft Configuration Optimization Including Optimized Flight Profiles. Recent Experiences in Multidisciplinary Analysis and Optimization. Part 1. J. Sobieski, ed., NASA CP–2327–PT–1, 1984, pp. 395–412.
17. Anderson, E., et al.: LAPACK Users' Guide. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
18. Vetterling, W.T., et al.: Numerical Recipes Example Book (C). Cambridge University Press, New York, 1988.
19. Rissanen, J.: Stochastic Complexity. J. Roy. Stat. B—Methodological, vol. 49, no. 3, 1997, pp. 223–239.

Table I.—Solution for the 122-operating-point, supersonic mixed-flow turbofan engine

Operating points			Optimum thrust, lb		Cascade improvement, percent
Number	Altitude, ft	Mach number	Cascade solution	Original NEPP solution	
20	Sea level	0.3	47 570	46 000	3.41
31	689	.3	19 119	18 690	2.30
57	10 000	.9	46 384	44 674	3.83
68	30 000	.9	14 566	13 827	5.34
85	36 089	1.1	10 080	9 857	2.26
100	40 000	1.5	14 403	14 311	.64
122	56 000	1.8	20 127	19 773	1.79

Table II.—Solution for the 47-operating-point, subsonic waverotor-topped engine

Operating points			Optimum thrust, lb		Cascade improvement, percent
Number	Altitude, ft	Mach number	Cascade solution	Original NEPP solution	
1	Sea level	0.25	70 075	70 075	0.0
13	10 000	.4	45 135	43 072	4.80
20	20 000	.51	29 279	29 221	.20
26	30 000	.60	18 827	17 898	5.19
32	30 000	.70	19 161	18 203	5.26
38	20 000	.80	30 898	30 110	2.62
47	40 000	.85	12 684	12 611	.58

Table III.—Benchmark solution with standard deviation and optimum thrust for the waverotor-topped engine

Operating point			Benchmark solution of optimum thrust		Approximate solutions of optimum thrust, lb						
					Regular optimization		Cascade optimization solutions				
					NLPQ		FD–SUMT–NLPQ		NLPQ–FD–NLPQ		NLPQ/NN– FD/Reg–NLPQ/NEPP
Number	Altitude, ft	Mach number	Average solution, lb	Standard deviation	Regression	Neural network (NN)	Regression	Neural Network (NN)	Regression	Neural network (NN)	
1	0	0.25	70 030	83	69 980	70 094	69 980	70 094	69 980	70 094	70 165
2	0	.10	80 220	48	80 188	80 271	80 188	80 271	80 188	80 271	80 238
3	0	0	88 827	55	88 794	88 818	88 794	88 818	88 794	88 818	88 751
4	5000	0	73 641	31	83 307	83 454	73 631	73 647	73 631	73 647	73 633
5	0	.10	80 029	377	80 207	80 289	80 207	80 289	80 207	80 289	80 318
6	5000	0.10	66 941	106	75 232	63 291	66 893	66 911	66 897	66 890	66 901
7	0	.20	72 963	899	73 052	73 156	73 052	73 156	73 052	73 156	73 079
8	5000	.20	61 703	101	68 357	67 675	61 647	61 664	61 645	61 664	61 641
9	0	.30	69 209	370	69 165	69 213	69 164	69 174	69 167	69 213	69 184
10	5000	.30	57 446	61	62 348	62 206	57 450	57 472	57 450	57 472	57 437

Table IV.—Good-quality input-output pairs for training the approximate methods

Operating point number	Number of good-quality input-output pairs	Bad quality input-output pairs			
		Zero thrust	Thrust difference exceeding 10 lb	Zero value for constraints	Constraint value difference exceeding 6 percent
1	2764	224	10	1	2
2	2768	223	7	2	1
3	2768	225	6	1	1
4	2764	235	2	0	0
5	2766	226	5	3	1
6	2763	237	0	1	0
7	2771	225	4	1	0
8	2763	235	0	1	2
9	2778	215	6	1	1
10	2772	227	2	0	0

Table V.—Error norm in thrust and constraints for a waverotor-topped engine

Operating point	Variation in thrust, percent		Auxiliary fan/compressor corrected speed (≤1.01)		Fan corrected speed (≤1.01)		Compressor corrected speed (≤1.01)	
	Regression	Neural network	Regression	Neural network	Regression	Neural network	Regression	Neural network
1	0.033	0.25	0.012	0.107	0.012	0.107	0.027	0.034
2	.036	.23	.017	.108	.017	.108	.032	.034
3	.036	.219	.018	.109	.018	.109	.032	.036
4	.015	.18	.013	.116	.013	.116	.048	.051
5	.037	.23	.017	.108	.017	.108	.032	.034
6	.017	.191	.014	.116	.014	.116	.05	.051
7	.034	.241	.013	.107	.013	.107	.031	.034
8	.018	.205	.017	.115	.017	.115	.051	.053
9	.032	.254	.012	.105	.012	.105	.024	.034
10	.025	.223	.022	.116	.022	.116	.043	.049

Operating point	Burner outlet temperature (≤3200 °R)		Auxiliary fan/compressor surge margin (≥15 percent)		Compressor surge margin (≥15 percent)	
	Regression	Neural network	Regression	Neural network	Regression	Neural network
1	0.037	0.08	0.073	0.478	1.238	0.994
2	.041	.084	.069	.426	1.169	1.034
3	.041	.085	.069	.419	1.127	.996
4	.022	.103	.044	.369	1.059	1.049
5	.041	.084	.071	.426	1.171	1.034
6	.022	.101	.046	.373	1.041	1.033
7	.039	.082	.069	.45	1.236	1.049
8	.023	.097	.053	.406	.993	.908
9	.036	.076	.074	.499	1.184	.937
10	.028	.09	.074	.473	.989	.775

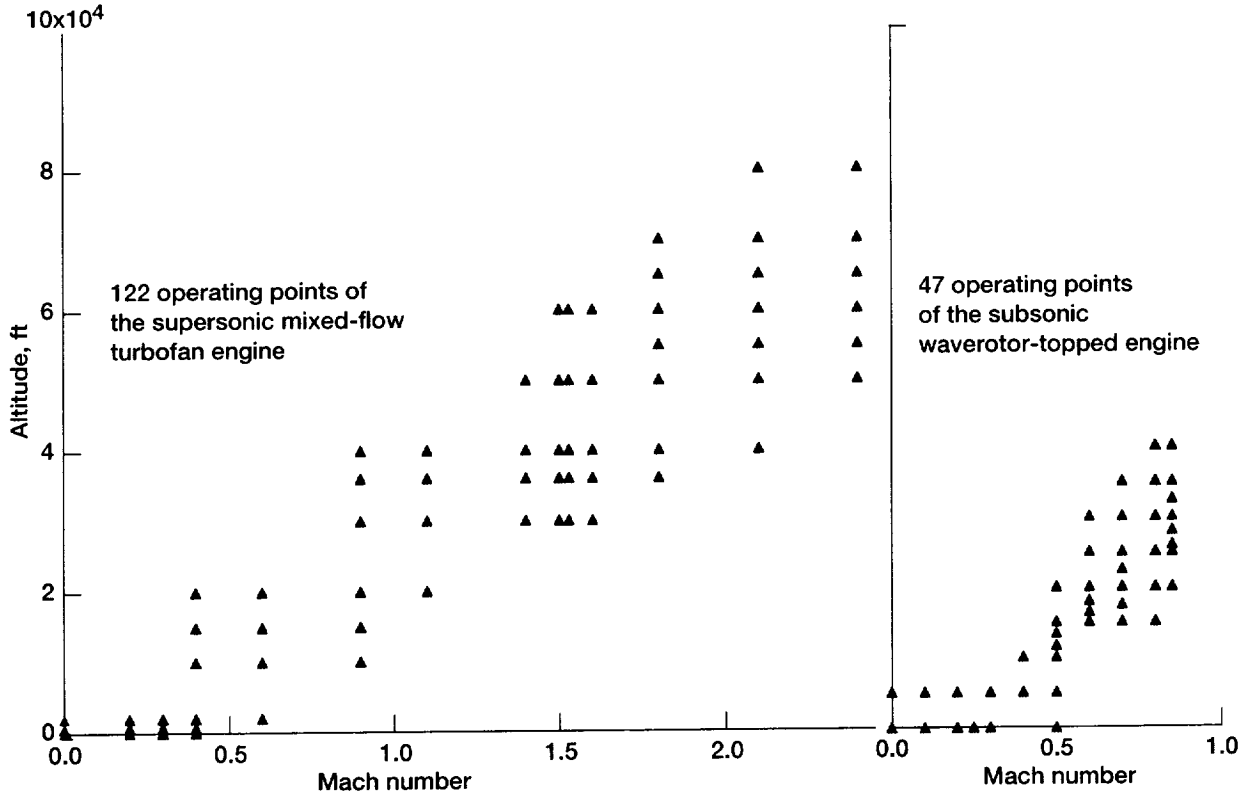

Figure 1.—Mission profile for supersonic and subsonic engines.

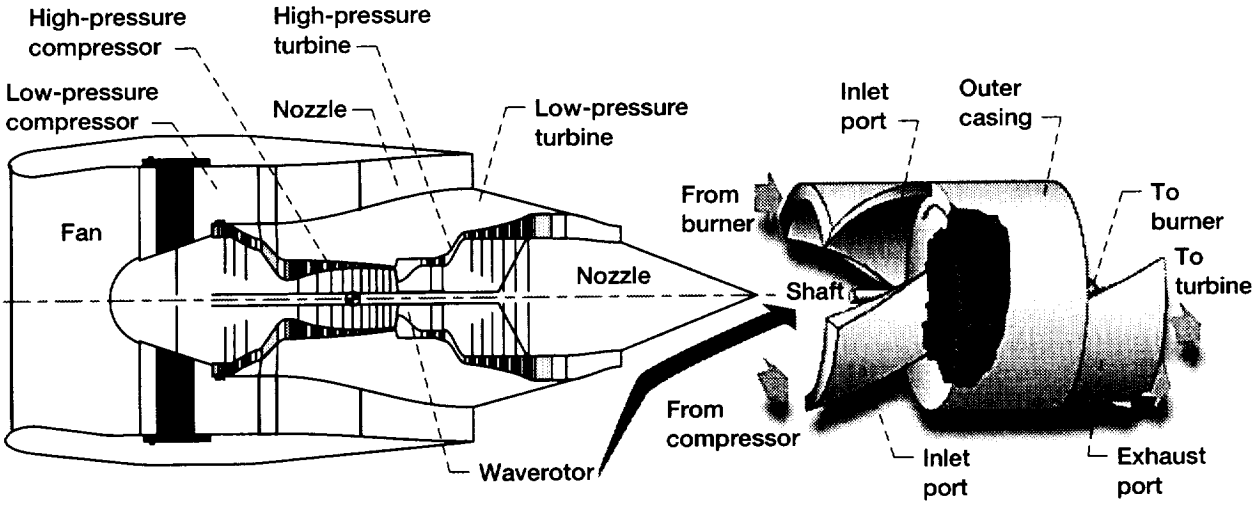

Figure 2.—Subsonic waverotor-topped gas turbine engine.

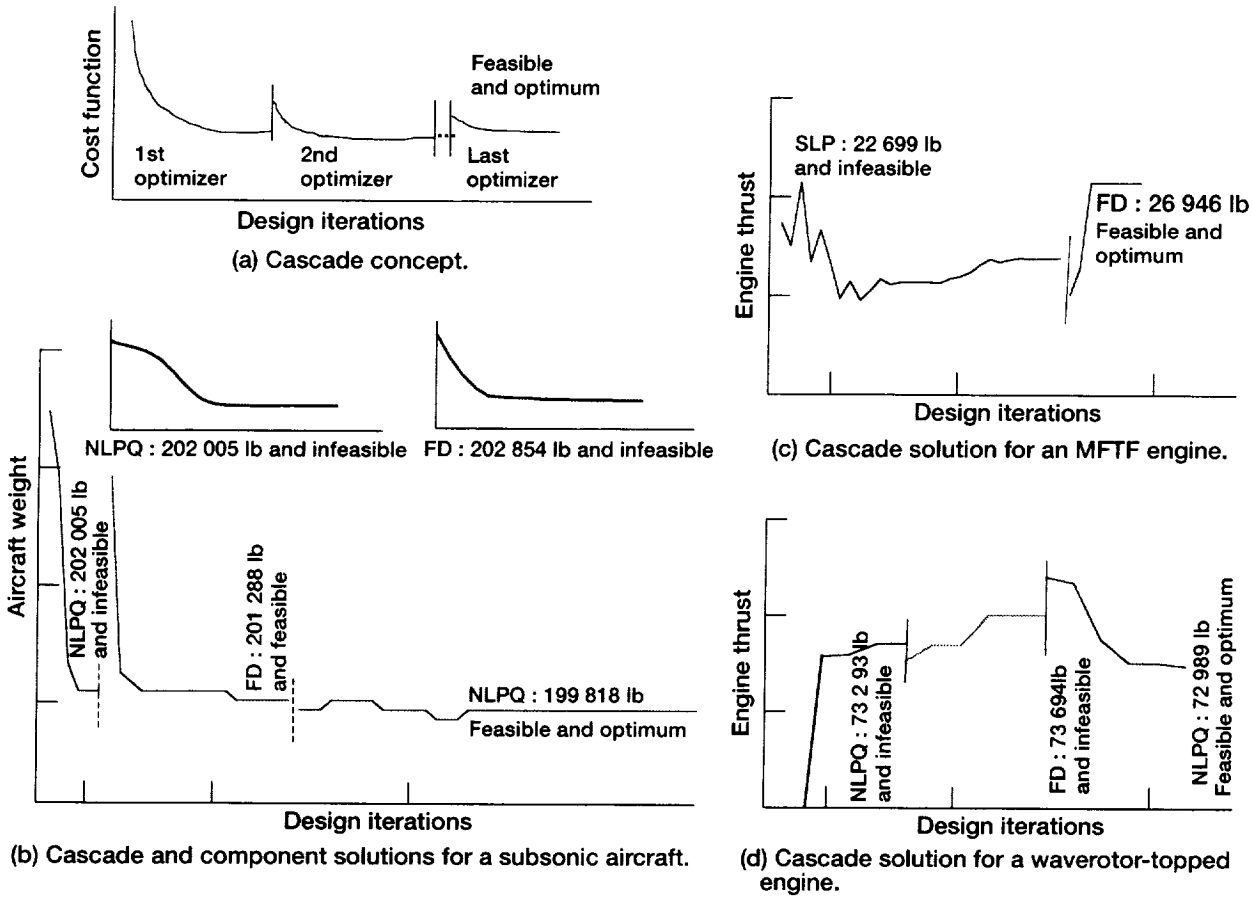

Figure 3.—Cascade solutions for aircraft and engines.

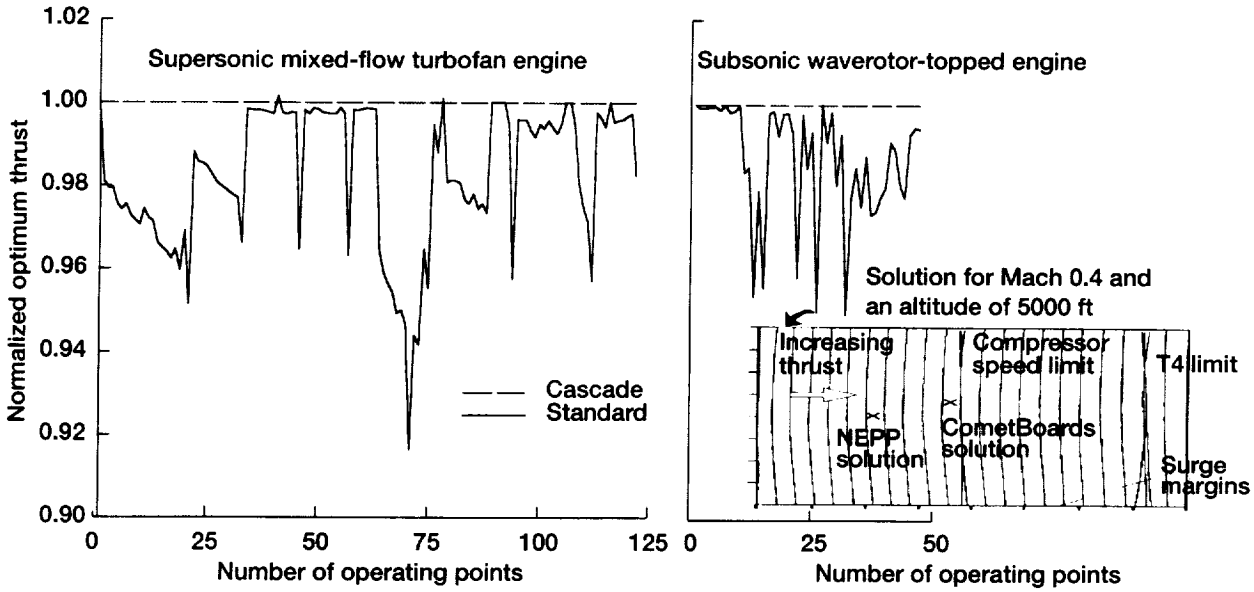

Figure 4.—Solutions for supersonic and subsonic engines, normalized with respect to cascade solutions.

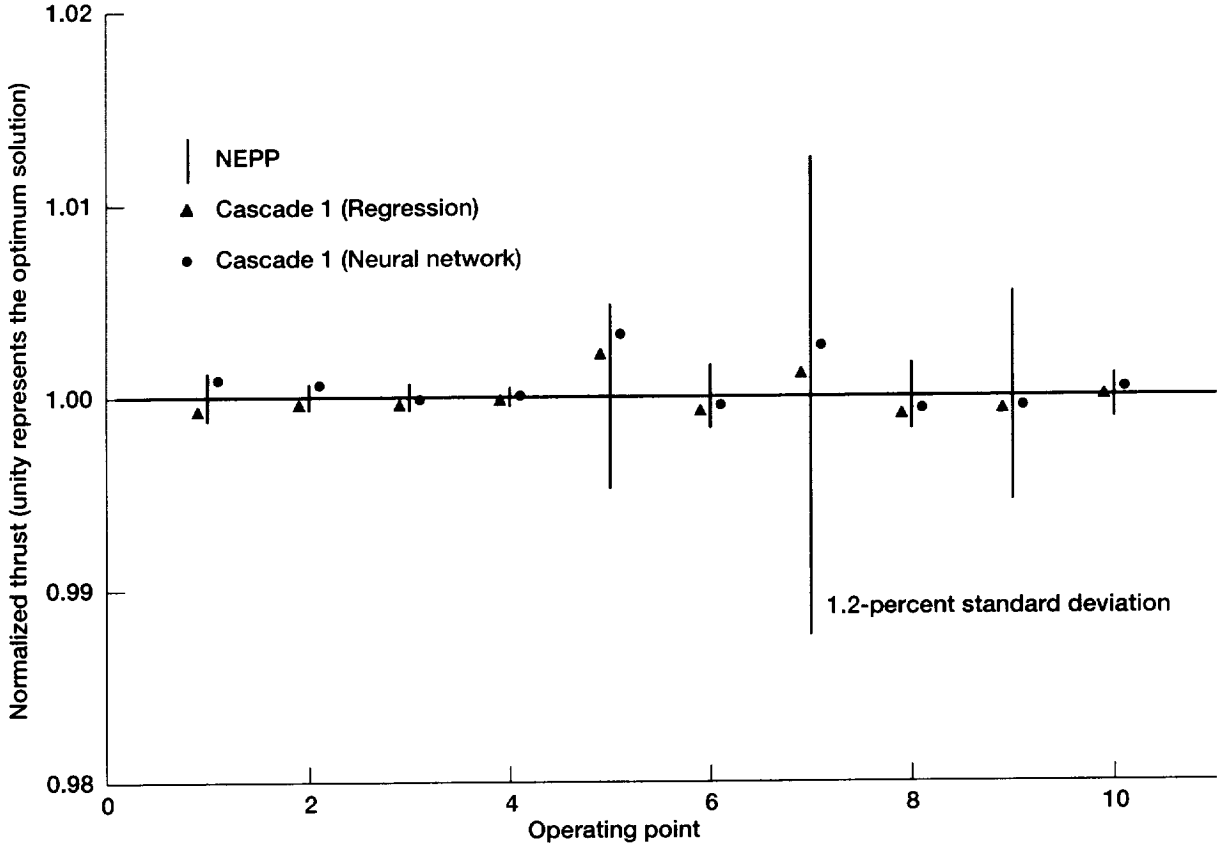

Figure 5.—Optimum solutions for a subsonic waverotor-topped engine at 10 operating points.

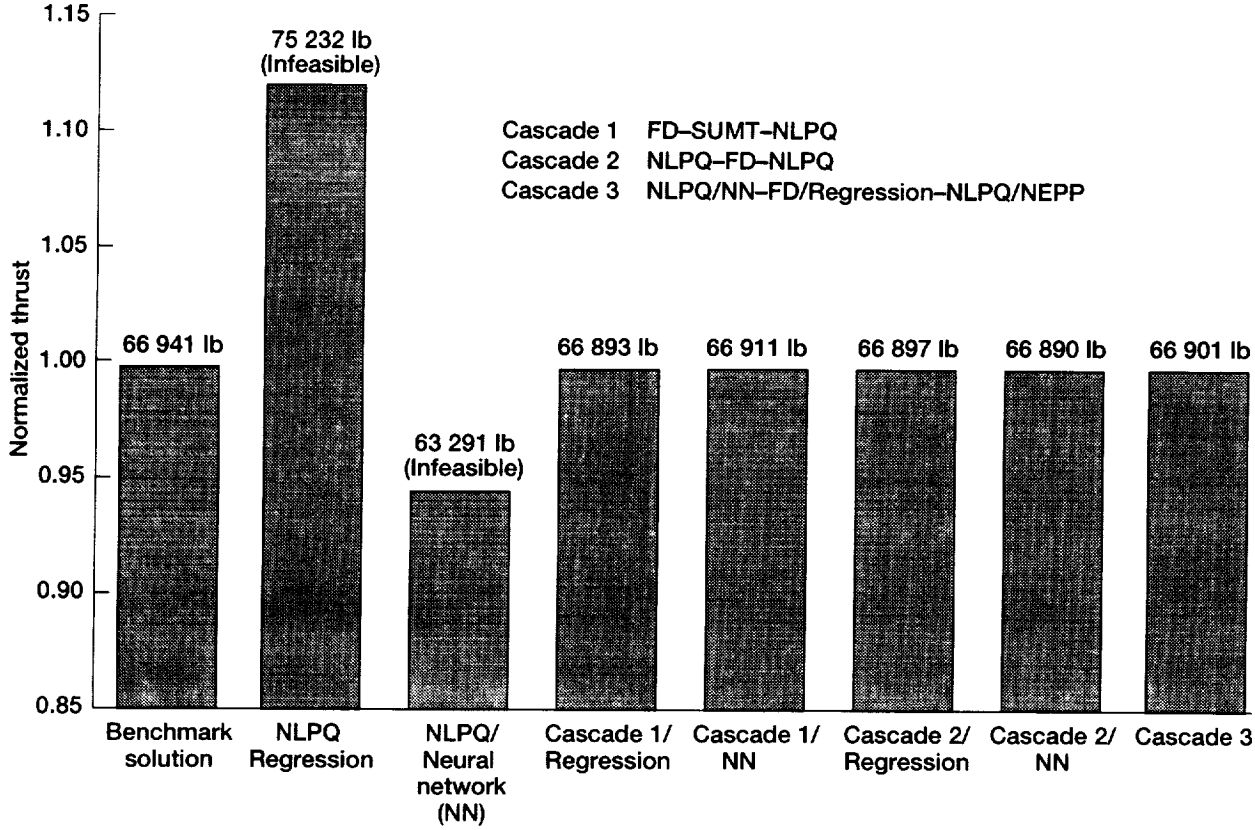

Figure 6.—Optimum thrust for a subsonic waverotor-topped engine at the sixth operating point.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 2000	Technical Memorandum

4. TITLE AND SUBTITLE

Cascade Optimization for Aircraft Engines With Regression and Neural Network Analysis-Approximators

5. FUNDING NUMBERS

WU-523-24-13-00

6. AUTHOR(S)

Surya N. Patnaik, James D. Guptill, Dale A. Hopkins and Thomas M. Lavelle

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
John H. Glenn Research Center at Lewis Field
Cleveland, Ohio 44135-3191

8. PERFORMING ORGANIZATION REPORT NUMBER

E-11686

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA TM—2000-209177

11. SUPPLEMENTARY NOTES

Surya N. Patnaik, Ohio Aerospace Institute, 22800 Cedar Point Road, Cleveland, Ohio 44142; James D. Guptill, Dale A. Hopkins, and Thomas M. Lavelle, NASA Glenn Research Center. Responsible person, Surya N. Patnaik, organization code 5930, (216) 433-5916.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Category: 39 Distribution: Nonstandard

This publication is available from the NASA Center for AeroSpace Information, (301) 621-0390.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

The NASA Engine Performance Program (NEPP) can configure and analyze almost any type of gas turbine engine that can be generated through the interconnection of a set of standard physical components. In addition, the code can optimize engine performance by changing adjustable variables under a set of constraints. However, for engine cycle problems at certain operating points, the NEPP code can encounter difficulties: nonconvergence in the currently available Powell's optimization algorithm and deficiencies in the Newton-Raphson solver during engine balancing. A project was undertaken to correct these deficiencies. Nonconvergence was avoided through a cascade optimization strategy, and deficiencies associated with engine balancing were eliminated through neural network and linear regression methods. An approximation-interspersed cascade strategy was used to optimize engine operation over the flight envelope. Replacement of Powell's algorithm by the cascade strategy improved the optimization segment of the NEPP code. The performance of the linear regression and neural network methods as alternative engine analyzers was found to be satisfactory. This report considers two examples—a supersonic mixed-flow turbofan engine and a subsonic waverotor-topped engine—to illustrate the results and discusses insights gained from the improved version of the NEPP code.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Optimization; Aircraft; Engine; Performance; Cascade strategy; Neural network; Regression method	24
	16. PRICE CODE: A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	